

★KOKZ T01 2002-580533/62 ★JP 2002198827-A
Recursive form structure convolutional code decoding method in data communication, involves multiplying code divisor polynomial to code sequence transmitted in selected state transition path

KOKUSAI DENKI KK 2000.12.27 2000JP-398260

U21 (2002.07.12) H03M 13/41, G06F 11/10

Novelty: Selective comparison of a recursive form structure convolutional code is performed using obtained state-transition choice information. A code sequence transmitted in a selected state-transition path is multiplied with a code divisor polynomial of the convolutional code and the result is updated in a path memory (5).

Detailed Description: An INDEPENDENT CLAIM is included for recursive form structure convolutional code decoder.

Use: For decoding error correcting code such as recursive form structure convolutional code using Viterbi decoder, in data communication application.

Advantage: Additional multiplication circuit of a code divisor polynomial is omitted, thereby reducing the circuit size and shortening arithmetic processing time.

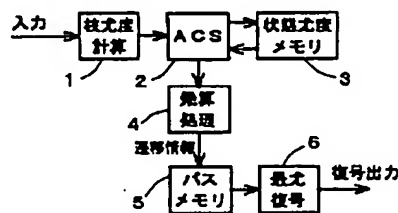
Description of Drawing(s): The figure shows the block diagram of the recursive form structure convolutional code decoder. (Drawing includes non-English language text).

Path memory 5

(9pp Dwg.No.1/8)

N2002-460835

T01-G01A1; U21-A06



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-198827

(P 2 0 0 2 - 1 9 8 8 2 7 A)

(43) 公開日 平成14年7月12日 (2002. 7. 12)

(51) Int. Cl. ⁷	識別記号	F I	テ-マコード (参考)
H03M 13/41		H03M 13/41	5B001
G06F 11/10	330	G06F 11/10	330 N 5J065

審査請求 未請求 請求項の数 9 O L (全9頁)

(21) 出願番号 特願2000-398260 (P 2000-398260)

(22) 出願日 平成12年12月27日 (2000. 12. 27)

(71) 出願人 000001122

株式会社日立国際電気

東京都中野区東中野三丁目14番20号

(72) 発明者 大西 誠

東京都小平市御幸町32番地 株式会社日立

国際電気小金井工場内

Fターム(参考) 5B001 AA10 AB02 AD06

5J065 AD10 AG05 AH02 AH03 AH06

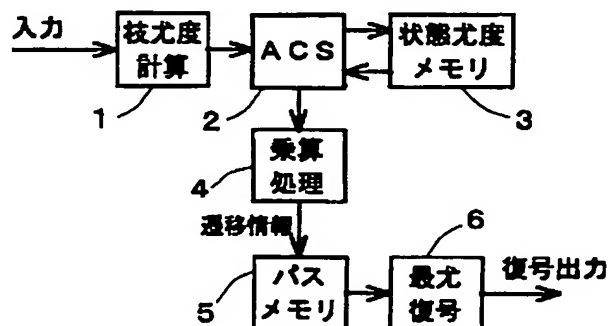
AH09 AH15 AH21 AH23

(54) 【発明の名称】 最尤復号方法及び最尤復号器

(57) 【要約】

【課題】 復号出力の軟判定尤度情報を得ることが可能な、再帰形組織畳み込み符号用の最尤復号器の構成法を提供する。

【解決手段】 従来の最尤復号器における加算比較選択処理部と、バスメモリの間に、再帰形組織畳み込み符号の除数多項式乗算処理部を設け、選択されたトレリス遷移バスに対応するビット系列に除数多項式乗算を行い、送信情報ビット系列を復号する。



【特許請求の範囲】

【請求項1】 k ビットの情報を n ビット（ただし、 n と k は正整数、かつ、 $n > k$ ）の符号に符号化する符号化率 r （ただし、 $r = k/n$ ）、拘束長 K の再帰形組織畳み込み符号を復号する最尤復号器において、前記再帰形組織畳み込み符号を加算比較選択演算し、得られる状態遷移選択情報によって、バスメモリ更新を行うとともに、前記再帰形組織畳み込み符号の符号除数多項式を乗積する演算を同時に実行し、乗積演算した値をバスメモリに記録することを特徴とする最尤復号方法。

【請求項2】 k ビットの情報を n ビット（ただし、 n と k は正整数、かつ、 $n > k$ ）の符号に符号化する符号化率 r （ただし、 $r = k/n$ ）、拘束長 K の再帰形組織畳み込み符号を復号する最尤復号器において、前記再帰形組織畳み込み符号を加算比較選択演算し、得られる状態遷移選択情報に基づき、遷移前状態の状態番号（ $0 \sim 2^{r-1} - 1$ ）と遷移後状態の状態番号（ $0 \sim 2^{r-1} - 1$ ）とから、選択した状態遷移パスに対応する送信符号列を構成し、該送信符号列に符号除数多項式を乗積して、該乗積結果をバスメモリに記録することを特徴とする最尤復号方法。

【請求項3】 請求項2記載の最尤復号方法において、前記状態番号 2^{r-1} と同じ数のバスメモリの記憶内容をMSB側に k ビットシフトして空いたLSB側の k ビットの位置に、前記符号除数多項式を乗積した結果得られる k ビットの情報を追加書き込みすることによって前記バスメモリの更新を行うことを特徴とする最尤復号方法。

【請求項4】 請求項1乃至請求項3のいずれかに記載の最尤復号方法において、状態尤度メモリから、現時点（ t_s ）の最尤状態（状態尤度の最も小さい状態、 m 番目の状態とする。但し、 $0 \leq m \leq 2^{r-1} - 1$ ）に至る一時点前（ t_{s-1} ）の $m_1 = m / 2$ 番目および $m_1 = m / 2 + 2^{r-1}$ 番目（但し、 $/$ は整数除算を示す）の状態の状態尤度 $ms(m_1)$ および $ms(m_2)$ を読み出し、 m_1 番目の状態から m 番目の状態への遷移に伴う枝尤度 $mb(m_1, m)$ と、 m_2 番目の状態から m 番目の状態への遷移に伴う枝尤度 $mb(m_2, m)$ を読み出し、軟判定出力尤度を次式によって計算し（但し、 $|x|$ は x の絶対値）、
$$mo = |ms(m_1) - ms(m_2) + mb(m_1, m) - mb(m_2, m)|$$
計算した前記軟判定出力尤度を記憶保持し、最尤復号出力が確定した時点で、前記保持された前記軟判定出力尤度を読み出して、軟判定出力尤度情報として出力することを特徴とする最尤復号方法。

【請求項5】 k ビットの情報を n ビット（ただし、 n と k は正整数、かつ、 $n > k$ ）の符号に符号化する符号

化率 r （ただし、 $r = k/n$ ）、拘束長 K の再帰形組織畳み込み符号を復号する最尤復号器であって、前記再帰形組織畳み込み符号を入力し、枝尤度を計算する枝尤度計算部と、

状態尤度を記憶する状態尤度メモリと、該状態尤度メモリに記憶された状態尤度と、前記枝尤度計算部が計算した枝尤度とによって状態遷移選択情報を求める加算比較選択演算部と、計算された該状態尤度に乗算処理を行う符号多項式乗算処理部と、

前記加算比較選択演算部から得られる前記状態遷移選択情報により、メモリ更新を行うバスメモリと、該バスメモリの出力を復号する最尤復号部とを有し、前記メモリの更新を行う時に、前記再帰形組織畳み込み符号の符号除数多項式を乗積する演算を同時に実行し、乗積した結果を前記バスメモリに記録することを特徴とする最尤復号器。

【請求項6】 k ビットの情報を n ビット（ただし、 n と k は正整数、かつ、 $n > k$ ）の符号に符号化する符号化率 r （ただし、 $r = k/n$ ）、拘束長 K の再帰形組織畳み込み符号を復号する最尤復号器において、最尤復号器の加算比較選択演算部から得られる状態遷移選択情報により、バスメモリ更新を行う処理段階で、前記再帰形組織畳み込み符号の符号除数多項式を乗積する演算を同時に実行し、乗積した結果をバスメモリに記録することを特徴とする最尤復号器。

【請求項7】 請求項5または請求項6のいずれかに記載の最尤復号器において、前記加算比較選択演算部で選択した前記状態遷移情報に基づき、遷移前状態の状態番号（ $0 \sim 2^{r-1} - 1$ ）と遷移後状態の状態番号（ $0 \sim 2^{r-1} - 1$ ）とから、選択した状態遷移パスに対応する送信符号列を構成し、該送信符号列に前記符号除数多項式を乗積し、該乗積結果を前記バスメモリに記録することを特徴とする最尤復号器。

【請求項8】 請求項5乃至請求項7のいずれかに記載の最尤復号器において、状態数 2^{r-1} と同じ数のバスメモリを設け、該バスメモリの記憶内容をMSB側に k ビットシフトして空いたLSB側 k ビットの位置に、前記符号除数多項式を乗積した結果得られる k ビットの情報を追加書き込みすることによってバスメモリ更新を行うことを特徴とする最尤復号器。

【請求項9】 請求項5乃至請求項8のいずれかに記載の最尤復号器において、軟判定尤度出力部と、出力尤度メモリとを設け、前記最尤復号器の状態尤度メモリから、現時点（ t_s ）の最尤状態（状態尤度の最も小さい状態、 m 番目の状態とする。但し、 $0 \leq m \leq 2^{r-1} - 1$ ）に至る一時点前

3

(t_{i-1}) の $m_1 = m/2$ 番目および $m_2 = m/2 + 2^{i-1}$ 番目 (但し、 $/$ は整数除算を示す) の状態の状態尤度 $m s (m_1)$ および $m s (m_2)$ を読み出し、

前記最尤復号器の枝尤度計算部から、 m_1 番目の状態から m 番目の状態への遷移に伴う枝尤度 $m b (m_1, m)$ と、 m_2 番目の状態から m 番目の状態への遷移に伴う枝尤度 $m b (m_2, m)$ を読み出し、

前記軟判定尤度出力部において、軟判定出力尤度 $m o = |m s (m_1) - m s (m_2) + m b (m_1, m) - m b (m_2, m)|$

を計算 (但し、 $|x|$ は x の絶対値) し、

前記出力尤度メモリに記憶保持し、

前記最尤復号器の最尤復号部によって、最尤復号出力が確定した時点で、該出力尤度メモリから読み出して、軟判定出力尤度情報として出力することを特徴とする最尤復号器。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、誤り訂正符号の復号器に係わり、特に再帰形組織畳み込み符号の復号に効果のある最尤復号器に関する。

【0002】

【従来の技術】データ伝送の信頼性を高めるための技術として、伝送路で生ずる伝送データの符号誤りを受信側で訂正する誤り訂正技術がある。その中でも、畳み込み符号を復号するビタビ復号器は、最尤復号 (受信符号に対して最も確からしい送信符号を推定し出力する復号法) が可能であり、多方面に適用されている。さらに近年、符号誤り訂正能力をより大きくできるターボ符号の技術が開発され、話題となっている。このターボ符号には、再帰形組織畳み込み符号という符号が用いられる。図面を用いて、これら畳み込み符号器について説明する。

【0003】図2は、従来の最尤復号器の動作を説明するための図である。図2aは、(非組織) 畳み込み符号器、図2bは、再帰形組織畳み込み符号器の構成を示すブロック図である。20は最尤復号器、21～25は排他論理和ゲート、26～28は1ビットの遅延素子、29は3ビットのシフトレジスタである。図2aに示す畳み込み符号器では、入力信号系列 x がシフトレジスタ29に入力され、0～3ビット遅延された信号 x_0 、 x_1 、 x_2 、 x_3 が出力される。排他論理和ゲート21と22により、 x_0 、 x_1 、 x_2 の排他論理和が演算され、符号 g_0 として出力される。他方、符号 g_1 は排他論理和ゲート23～25により、 x_0 、 x_1 、 x_2 、 x_3 の排他論理和から計算される。信号系列 x の N ビット遅延を x の N 乗 (x^N) で表し、排他論理和を $+$ で表現すると、符号 g_0 および g_1 は、次の式(1)に示す符号多項式によって表される (但し、 $x^0 = 1$ 、 $x^1 = x$ と表す)。

【数1】

$$\left. \begin{aligned} g_0 &= 1 + x + x^2 + x^3 \\ g_1 &= 1 + x + x^2 + x^3 \end{aligned} \right\} \dots\dots\dots \text{式(1)}$$

図2aの畳み込み符号器は、入力信号系列 x と、式(1)の符号多項式を掛け算することになるので、符号多項式乗算回路とも呼ばれる。

【0004】図2bに示す再帰形組織畳み込み符号器

は、図2aの畳み込み符号器と、 g_0 側の回路構成が異なっている。図2bの信号 u は、

$$u = x + u_1 + u_2$$

と表すことができる。 u のべき乗を移項して、

$$u + u_1 + u_2 = u \times (1 + x + x^3) = x$$

とし (排他論理和では引き算も $+$ で表される)、 u について解くと、

$$u = x / (1 + x + x^3) = x / g_0 \dots\dots\dots \text{式(2)}$$

となり、図2bの上半分の回路は式(1)の符号多項式 g_0 によって割り算を行う除算回路であることがわかる。こうして、図2bの出力は、式(2)の信号 u に、さらに式(1)の符号多項式 g_1 を掛け算して、 $x \times (g_1 / g_0)$ となることが分かる。

【0005】図2aの畳み込み符号器の符号生成多項式は (g_0 、 g_1) で表されるのに対し、図2bの再帰形組織畳み込み符号器による生成有理式は (1 、 g_1 / g_0) となる。両者は同じ符号多項式の係数を持っているので、生成される符号は異なるが、特性は同じであり、全く同じ構造のビタビ復号器 (最尤復号器) により復号することができる。但し、最尤復号器は符号器シフトレジスタのビット列を復号するので、再帰形組織畳み込み符号を復号する場合には、符号器シフトレジスタのビット系列 (u) が復号される。したがって入力信号系列

(x) を再生するには、図2cに示すように、 g_0 を u に掛け算する乗算回路が必要となる。図2cは、再帰形組織畳み込み符号を復号する場合の基本的回路構成を示すブロック図である。図2に示すように、最尤復号器20の出力 u の後段に乗算回路を付加した構成となる。

【0006】次に、畳み込み符号を復号する最尤復号器について以下に説明する。まず、最尤復号において重要な役割をする畳み込み符号器の状態の概念と、トレリス線図について図3を用いて説明する。図3は、最尤復号器の動作を説明するための畳み込み符号のトレリス線図である。図2aに示す畳み込み符号器は、入力データ x_0 から符号化出力 g_0 と g_1 を生成し出力する。この畳み込み符号器は、入力1ビットに対して2ビット出力されるので、符号化率 $r = 1/2$ 、また、入力データ4ビットで符号を生成するので、拘束長 $K = 4$ の畳み込み符号器と呼ばれている。

【0007】符号器のシフトレジスタが保持している3ビット x_0 、 x_1 、 x_2 を符号器の状態という。3ビットであるから000～111の8状態があり、入力データが入

る度に8状態が遷移しながら符号化が行われる。この状態遷移の状況を図に表したものをトレリス線図と呼ぶ。図3aは、時点 t_{i-1} から時点 t_i への遷移状況を示すトレリス線図である。また、図3bは、トレリスの基本単位を示す図である。図3aにおいて、状態番号 x_0, x_1, x_2 の状態に x_0 が入力され、同時に x_2 が抜け出ていき、状態番号 x_1, x_2, x_0 の状態に移る。この時、 $m = x_1, x_2$ ($= 0 \sim 3$) とすると、 m ($x_0 = 0$) および $m + 4$ ($x_0 = 1$) の状態から、 $2m$ ($x_0 = 0$) および $2m + 1$ ($x_0 = 1$) の状態へ遷移する。すなわち、図3aの8状態トレリスは、図3bの基本単位トレリスが4個 組合わさったものである。符号化出力 g_0 と g_1 は、符号 g_0 と g_1 の生成多項式(1)で計算される。図3bの状態遷移に伴って出力される符号 g_0 と g_1 は、式(1)に図3bの x, m, mx_0 ($m = x_1, x_2$) を代入して式(3)に示すようになる。但し、式(3)において x' は x の論理反転を示す。

【数2】

$$\left. \begin{array}{l} (x_2, x_0 = 00, 11 \text{ のとき}) \\ \quad g_0 = x_1 \\ \quad g_1 = x_1 + x_2 \\ (x_2, x_0 = 01, 10 \text{ のとき}) \\ \quad g_0' = x_1' \\ \quad g_1' = (x_1 + x_2)' \end{array} \right\} \dots\dots\dots \text{式(3)}$$

【0008】以上述べたように、畳み込み符号のトレリス構造(遷移前後の状態番号、および出力される符号)は、状態番号 $m = 0 \sim 2^{k-1} - 1$ を与えると、一意的に決定される。このトレリス構造を用いて畳み込み符号の最尤復号を行うのが、ビタビアルゴリズムによるビタビ復号器である。図4によって、従来の再帰形組織畳み込み符号の復号器の動作を説明する。図4は、最尤復号器を用いて構成した再帰形組織畳み込み符号の復号器の従

$$\left. \begin{array}{l} mb(y_0, y_1, g_0, g_1) \\ \quad = d_H(r_0, g_0) + d_H(r_1, g_1) \quad \text{硬判定} \\ \quad = ((y_0 - I)^2 + (y_1 - Q)^2)^{1/2} \quad \text{軟判定} \\ \text{または} \quad |y_0 - I| + |y_1 - Q| \quad \text{軟判定} \end{array} \right\} \dots\dots\dots \text{式(4)}$$

式(4)において、 $d_H(x, y)$ は、符号 x, y のハミング距離を表す。

【0012】実際の最尤復号器では、図4に示した枝尤度計算部1で、受信符号が入力される毎に全ての遷移枝の枝尤度を計算する。求めた枝尤度を用いて、加算比較選択(ACS)演算部2で状態尤度を計算する。状態尤度は各状態に至る遷移パスの枝尤度を全て加算したものである。実際には図3bのトレリス線図で示されているように、各状態には2本ずつ遷移枝が入っているの、それぞれの遷移枝の枝尤度を1状態前の状態尤度に加算

来のブロック構成図である。図4において、40は(非組織畳み込み符号の)最尤復号器であり、要素1~6で構成される。1は枝尤度計算部、2は加算比較選択演算部(ACS演算部: Add, Compare, Select演算部)、3は状態尤度メモリ、5はバスメモリ、6は最尤復号部である。さらに41は、図2で説明した(符号多項式)乗算回路である。

【0009】図4において、最尤復号器40では、時点 t_i の全ての状態に至る遷移パス(図2aの例では8本)を候補パスとして保持しておき、受信した符号 r_0, r_1 を手がかりとして送信機の符号器の状態遷移として最も確からしい遷移パスを選択(最尤選択)することで符号器の状態遷移を推定しながら復号を行う。ここで確からしさを具体的に表す量として尤度を用いる。そして、状態番号 x, m から mx_0 の状態遷移に伴って出力される符号 g_0, g_1 と、実際に受信した符号 r_0, r_1 とのハミング距離を、その遷移枝の枝尤度とする。

【0010】受信符号 r_0, r_1 は受信信号 y_0, y_1 を識別して、

20 $y > 0$ のときは $r = 0$

$y < 0$ のときは $r = 1$ とする。

このような0、1判定した復号値で求める尤度を硬判定尤度という。

【0011】他方、状態遷移に伴う符号(g_0, g_1) = (0, 0)、(0, 1)、(1, 0)、(1, 1)を直交信号点座標(I, Q) = (1, 1)、(-1, 1)、(1, -1)、(-1, -1)に割り当て、直交信号点(I, Q)と受信信号点(y_0, y_1)とのユークリッド距離、または直交、同相成分の差の絶対値和を尤度として与える方法もある。この場合には、求めた尤度(実数値)を数ビットの整数値に丸めて尤度とする。こうして求めた枝尤度を軟判定尤度という。従って、枝尤度は式(4)で与えられる。

【数3】

(Add) し、尤度を比較(Compare) し、確からしい方の遷移パスを選択(Select) する。

【0013】枝尤度をハミング距離で表しているの、状態尤度も小さい方が確からしい(尤度が高い) ことになる。ACS演算部2が計算した状態尤度を、状態尤度メモリ3に格納するとともに、ACS演算によって選択した遷移情報をバスメモリ5に送る。バスメモリ5では全ての状態に至る遷移パスを記憶しておく。データの入力終了した時点で状態尤度の最も高い(数値としては最も小さい) 状態に至る遷移パスを符号器の遷移パスとして

推定し、最尤復号部 6 で最尤復号して (符号多項式) 乗算回路 41 へ出力する。乗算回路 41 は最尤復号結果に数 1 の g_0 を掛け算し、再帰形組織畳み込み符号の復号を行う。

【0014】上記のように、再帰形組織畳み込み符号は (通常の) 畳み込み符号の最尤復号器と乗算回路を組み合わせることにより復号することができる。しかしながら、軟判定情報を出力する最尤復号器では、この従来方法では再帰形組織畳み込み符号の復号はできない。図 5 を用いて、軟判定尤度情報を出力する最尤復号器について説明する。図 5 は、トレリス遷移と生き残りパスの説明するための図である。符号の尤度は、符号間距離で表わされる。畳み込み符号の符号間距離は、1 ビット異なる情報系列を符号化したときに得られる符号のハミング距離で表わすことができる。例えば、 $t = t_1$ の時点における状態 0 (状態尤度 m_{s_0}) から、情報系列 $x_1 = (0, 0, 0, 0)$ を符号化したときのトレリスを A で示し、情報系列 $x_2 = (1, 0, 0, 0)$ を符号化したときのトレリスを B で示す。図 5 において、情報系列 x_1 に対しては符号系列 $g_1 = (00, 00, 00, 00)$ が出力され、 $t = t_2$ の時点で状態尤度は $m_{s_1} = m_{s_0}$ となる。他方、トレリス B に対しては、符号系列は $g_2 = (11, 01,$

10

20

$$m_o = |m_s(m_1) - m_s(m_2) + m_b(m_1, m_2) - m_b(m_2, m_1)| \cdots \text{式(5)}$$

【0016】式(5)から分かるように、復号出力の軟判定尤度は最尤状態の状態尤度から求められ、復号出力そのものとは関係しない。そのため、再帰形組織畳み込み符号の復号出力は、符号多項式の乗算回路により求められるが、符号多項式を乗算する前の復号系列の軟判定尤度を荷重加算しても、再帰形組織畳み込み符号の復号出力の軟判定尤度は得られない。さらには、符号多項式乗算回路は、硬判定 (“0、1” の 2 値論理) 出力の場合には論理回路による構成が可能であるが、軟判定出力の場合には多値論理となるので、通常の (多値の) 乗算回路、加算回路が必要となり、回路規模、演算処理時間とも増加する。

【0017】

【発明が解決しようとする課題】 前述の従来技術による再帰形組織畳み込み符号の最尤復号器の構成には、以下に示す欠点がある。

(1) (復号出力の信頼度としての) 軟判定尤度情報を (符号多項式) 乗算回路によって処理する場合の荷重係数が定められない。

(2) 軟判定尤度情報を処理する演算回路の回路規模、演算処理時間が大きくなり、簡単な回路構成では処理できない。

本発明の目的は、上記のような欠点を除去するためのものである。

【0018】

【課題を解決するための手段】 上記の目的を達成するために、本発明の最尤復号器は、最尤復号器に乗算回路を

11, 11) となり、 $t = t_2$ の時点で状態尤度は $m_{s_1} = m_{s_0} + 7$ となる。ここで、1 ビット異なる情報系列 x_1 、 x_2 に対して、符号系列 g_1 、 g_2 は 7 ビット異なっており、これが畳み込み符号の最小自由距離と言われるものである。

【0015】 $t = t_2$ の時点でトレリス A とトレリス B は状態 0 に遷移するが、ACS 演算により、状態尤度の小さいトレリス A ($m_{s_1} < m_{s_2}$) の方が選択されて生き残り、トレリス B は A に合流し捨てられる。こうして、全ての状態に至る生き残りパスの中で、状態尤度の最小のパスが最尤パスである。ビタビ復号器の最尤復号出力は、その時点の最尤パスによって決定される。今、 $t = t_2$ の時点で、状態 5 が最尤状態 (状態尤度の最も小さい状態) と判定され、トレリス C が捨てられ、トレリス D が生き残り最尤パスとして選択されたとする。このとき、最尤復号出力に対する尤度は、トレリス C とトレリス D の状態尤度の差となる。即ち、復号出力の尤度 (m_o) は最尤状態 ($m_L = 5$) に遷移する 1 時点前の 2 つの状態尤度 ($m_s(m_1)$ 、 $m_s(m_2)$ 、 $m_1 = m_L / 2$ 、 $m_2 = m_L / 2 + 2^{t-1}$ 、 $/$ は整数除算を表わす。) と、その遷移に伴う 2 つの枝尤度 ($m_b(m_1, m_L)$ 、 $m_b(m_2, m_L)$) から、次の式(5)で与えられる。

従属接続せず、最尤復号器の中で符号多項式乗算処理を行う。即ち、最尤復号器 ACS 演算部は、ACS 演算により選択された遷移情報をバスメモリに送り、バスメモリは選択された遷移パスを記憶する。この遷移パスに基づいて最尤復号が行われるので、初めから符号多項式乗算処理を行ったパスをバスメモリに記憶しておくものである。

【0019】また、本発明の最尤復号方法は、 k ビットの情報を n ビット (ただし、 n と k は正整数、かつ、 $n > k$) の符号に符号化する符号化率 r (ただし、 $r = k / n$)、拘束長 K の再帰形組織畳み込み符号を復号する最尤復号器において、再帰形組織畳み込み符号を加算比較選択演算し、得られる状態遷移選択情報によって、バスメモリ更新を行うとともに、再帰形組織畳み込み符号の符号除数多項式を乗積する演算を同時に実行し、乗積演算した値をバスメモリに記録するものである。

【0020】また、本発明の最尤復号方法は、再帰形組織畳み込み符号を加算比較選択演算し、得られる状態遷移選択情報に基づき、遷移前状態の状態番号 ($0 \sim 2^{t-1} - 1$) と遷移後状態の状態番号 ($0 \sim 2^{t-1} - 1$) とから、選択した状態遷移パスに対応する送信符号列を構成し、送信符号列に符号除数多項式を乗積して、乗積結果をバスメモリに記録するものである。さらに、本発明の最尤復号方法は、状態番号 2^{t-1} と同じ数のバスメモリの記憶内容を MSB 側に k ビットシフトして空いた LSB 側の k ビットの位置に、符号除数多項式を乗積した結果得られる k ビットの情報を追加書き込みすることによってバスメモリの更新を行うものである。

50

【0021】また、本発明の最尤復号器は、再帰形組織畳み込み符号を入力し、枝尤度を計算する枝尤度計算部と、状態尤度を記憶する状態尤度メモリと、状態尤度メモリに記憶された状態尤度と、枝尤度計算部が計算した枝尤度とによって状態遷移選択情報を求める加算比較選択演算部と、計算された状態尤度に乗算処理を行う符号多項式乗算処理部と、加算比較選択演算部から得られる状態遷移選択情報により、メモリ更新を行うバスメモリと、バスメモリの出力を復号する最尤復号部とを有し、メモリの更新を行う時に、再帰形組織畳み込み符号の符号除数多項式を乗積する演算を同時に実行し、乗積した結果をバスメモリに記録するものである。

【0022】また、本発明の最尤復号器は、 k ビットの情報を n ビット（ただし、 n と k は正整数、かつ、 $n > k$ ）の符号に符号化する符号化率 r （ただし、 $r = k/n$ ）、拘束長 K の再帰形組織畳み込み符号を復号する最尤復号器において、最尤復号器の加算比較選択演算部から得られる状態遷移選択情報により、バスメモリ更新を行う処理段階で、再帰形組織畳み込み符号の符号除数多項式を乗積する演算を同時に実行し、乗積した結果をバスメモリに記録するものである。さらに、本発明の最尤復号器は、加算比較選択演算部で選択した状態遷移情報に基づき、遷移前状態の状態番号（ $0 \sim 2^{r-1} - 1$ ）と遷移後状態の状態番号（ $0 \sim 2^{r-1} - 1$ ）とから、選択した状態遷移パスに対応する送信符号列を構成し、送信符号列に符号除数多項式を乗積し、乗積結果を前記バスメモリに記録するものである。

【0023】さらに、本発明の最尤復号器は、状態数 2^{r-1} と同じ数のバスメモリを設け、バスメモリの記憶内容をMSB側に k ビットシフトして空いたLSB側 k ビットの位置に、符号除数多項式を乗積した結果得られる k ビットの情報を追加書き込みすることによってバスメモリ更新を行う。またさらに、本発明の最尤復号器は、軟判定尤度出力部と、出力尤度メモリとを設け、最尤復号器の状態尤度メモリから、現時点（ t_s ）の最尤状態（状態尤度の最も小さい状態、 m 番目の状態とする。但し、 $0 \leq m \leq 2^{r-1} - 1$ ）に至る一時点前（ t_{s-1} ）の $m_1 = m/2$ 番目および $m_2 = m/2 + 2^{r-1}$ 番目（但し、 $/$ は整数除算を示す）の状態の状態尤度 $m_s(m_1)$ および $m_s(m_2)$ を読み出し、最尤復号器の枝尤度計算部から、 m_1 番目の状態から m 番目の状態への遷移に伴う枝尤度 $m_b(m_1, m)$ と、 m_2 番目の状態から m 番目の状態への遷移に伴う枝尤度 $m_b(m_2, m)$ を読み出し、軟判定尤度出力部において、軟判定出力尤度 $m_o = |m_s(m_1) - m_s(m_2)| + m_b(m_1, m) - m_b(m_2, m)|$

を計算（但し、 $|x|$ は x の絶対値）し、出力尤度メモリに記憶保持し、最尤復号器の最尤復号部によって、最尤復号出力が確定した時点で、出力尤度メモリから読み出して、軟判定出力尤度情報として出力するものであ

る。

【0024】

【発明の実施の形態】本発明の最尤復号器は、最尤復号器と乗算回路とを従属接続せず、最尤復号器の中で符号多項式乗算処理を行うために、図4に示した最尤復号器40のACS演算部2は、ACS演算により選択された遷移情報をバスメモリ5に送り、バスメモリ5は選択された遷移パスを記憶しておく。最尤復号はこの遷移パスに基づいて行われるので、初めから符号多項式乗算処理を行ったパスをバスメモリ5に記憶しておけば、目的が達せられる。

【0025】このため、バスメモリの構成に若干の変更を行う。通常、バスメモリにはACS演算によって得られたパスの遷移情報（現時点の最尤状態に遷移した一時点前の状態番号）が記憶される。しかしながら、この方法では上述したバスメモリの打ち切り長だけ遡って（トレースバック処理）最尤復号をする必要がある。この欠点を回避し、任意の生残りパスに対する復号結果を容易に出力することができるバスメモリ構成法が、平成5年特許公開第315976号公報「ヴィタビ復号器」に述べられている。

【0026】このバスメモリ構成法を、図6を用いて説明する。図6において、61と63はバスメモリ、62はバス演算回路、610～612と630～632はバスメモリ内の1ワード記憶領域である。バスメモリ61と62は、状態数 2^{r-1} に等しいワード数の記憶領域を有し、各ワードはバスメモリ打ち切り長に等しいビット数のデータを記憶できるものとする。バスメモリ61と62はACS演算部から得られる遷移情報（状態遷移前後の状態番号 $m_0 \rightarrow m_1$ ）を基に、各状態に至る生残りパスを記憶する。

【0027】まず遷移前のパスを記憶している第1のバスメモリ63から m_0 番目のパス情報を読み出し、バス演算回路62でMSB側に1ビットシフトする。次に遷移後の状態番号 m_1 のLSBビットをバス情報のLSBに格納する。得られた新しいバス情報は、遷移後のパスを記憶する第2のバスメモリ61の、 m_1 番目のバス情報として書き込む。こうして得られたバス情報には、各時点での状態番号のLSBビットが時系列的に並べられている。

【0028】図3bの説明において、遷移後の状態番号（ $x_1 x_0$ ）のLSBビット（ x_0 ）は、その状態遷移が符号器で起こったとしたときの符号器の入力ビットであり、それは即、その符号を受信したときの復号ビットである。従って、図6のバスメモリに記憶されているバス情報は各生残りパスに対する復号出力である。図6の構造のバスメモリでは、遷移後の状態番号のLSBビットを記憶しているが、これを符号多項式乗算回路で処理した結果に置き換えれば、本発明の目的が達せられる。

【0029】図7を用いて、その方法を具体的に説明する。図7aは図3bで説明した（非組織）畳み込み符号用

の最尤復号器のACS演算処理と同じものであるが、遷移枝には符号器の情報系列が付してある。図7aの(非組織)畳み込み符号の場合には、パスメモリに渡す遷移情報としては、遷移前の状態番号と、遷移後の状態番号のLSBビットである。パスメモリでは、遷移前の状態番号のアドレス位置にあるパスメモリの内容を、遷移後状態番号のアドレス位置のパスメモリに書きこみ、MSB側に1ビットシフトし、空いたLSBに遷移後状態番号のLSB1ビットを書きこむ。再帰形組織畳み込み符号の場合には、このビット列に図7bに示す g_0 乗算回路によって式(1)の $g_0 = 1 + x + x^3$ を掛けることで、符号器入力ビット列が復元できる。例えば、ビット列が図7bに示す $0x_1x_10$ (状態 $0x_1x_1$ から状態 x_1x_10 への遷移に伴う送信ビット列)の場合には、 x_1 が出力される。そこで、パスメモリには図7cに示すように、 x_1 (ビット列が $0x_1x_10$ 、または $1x_1x_11$ のとき)、または x_1' (x_1 の反転ビット:ビット列が $0x_1x_11$ 、または $1x_1x_10$ のとき)を出力すれば良い。こうして復元した符号器入力ビット列をパスメモリに書きこんでいけば、最尤復号器の出力として、符号多項式を乗算したビット列が得られ、再帰形組織畳み込み符号用の最尤復号器が構成できる。

【0030】次に、本発明による最尤復号器の実施の形態を、図1によって説明する。1は枝尤度計算部、2は加算比較選択(ACS)演算部、3は状態尤度メモリ、4は符号多項式乗算処理部、5はパスメモリ、6は最尤復号部である。図1の実施例は、図4の40に示した従来の最尤復号器を、構成要素1~3と、構成要素5、6に分割し、間に符号多項式乗算処理部4を挿入したものであり、構成要素1~3までの動作は図4と同様である。図1において、加算比較選択(ACS)演算部2はACS演算によって選択した遷移情報(遷移前の状態番号)を符号多項式乗算処理部4に送る。符号多項式乗算処理部4では図7で説明した処理を行う。すなわち、遷移前の状態番号($x_1x_1x_1$)と遷移後の状態番号($x_1x_1x_0$)から、選択された遷移枝に伴う符号器のシフトレジスタのビット列($x_1x_1x_1x_0$)を再生し、これに、図7bに示す符号除数多項式 g_0 の乗算回路によって、式(1)の $g_0 = 1 + x + x^3$ を掛ける。

【0031】この乗算結果は再帰形組織畳み込み符号器の入力情報ビットに一致し、最尤状態の乗算結果を選べば、再帰形組織畳み込み符号の最尤復号結果となる。この g_0 乗算結果と、遷移前の状態番号を遷移情報としてパスメモリ5に送る。パスメモリ5では遷移前の状態番号をアドレスとするパスメモリの内容を遷移後の状態番号のアドレス位置に書き込み、MSB側にシフトし、空いたLSB位置に上記した g_0 乗算結果を書き込む。こうして、パスメモリ5には、再帰形組織畳み込み符号の候補復号ビット系列が記憶され、最尤復号部6によって、最尤状態(状態尤度の最も小さい状態)のパスメモリか

ら、パスメモリの内容を読み出せば、再帰形組織畳み込み符号の最尤復号結果が得られる。

【0032】図1の本発明の実施例によれば、ACS演算部2からパスメモリ5に渡す遷移情報に乗算処理を施すことで、再帰形組織畳み込み符号の復号に必要な符号除数多項式の乗算処理が行えるので、従来の方法のように、(非組織)畳み込み符号の最尤復号器の後段に乗算回路を接続する必要が無く、処理回路の縮減、演算時間の短縮化が図れる。また乗算処理部4を挿入した構成と、外した構成とを切り替え可能とすれば、(非組織)畳み込み符号にも、再帰形組織畳み込み符号にも対応できる最尤復号器が構成できる。

【0033】図1の実施例では、復号出力は硬判定で行われる。すなわち、復号出力の信頼度を表す軟判定尤度出力は考慮していないので、従来の方法でも実現可能である。そこで、本発明による第2の実施例として、軟判定尤度出力の得られる再帰形組織畳み込み符号用の最尤復号器の実施例を図8に示す。1は枝尤度計算部、2は加算比較選択(ACS)演算部、3は状態尤度メモリ、4は符号多項式乗算処理部、5はパスメモリ、6は最尤復号部、80は出力尤度演算部、81は出力尤度メモリである。図8の実施例は、図1の本発明による最尤復号器の実施例1に、出力尤度演算部80と、出力尤度メモリ81を設けたものであり、構成要素1~6の動作は図1の実施例と同様である。

【0034】図8において、出力尤度演算部80は状態尤度メモリ3から、状態遷移後の最尤状態(m 番目の状態とする)に遷移する状態遷移前の状態尤度 $m_s(m_1)$ と、ACS演算によって選択されなかった状態尤度 $m_s(m_2)$ ($m_1, m_2 = m/2$ または $m/2 + 2^{i-1}$ 。但し、 i は整数除算を表す)を読み出し、さらに、枝尤度計算部1より、 m_1 番目の状態から m 番目の状態への遷移に伴う枝尤度 $m_b(m_1, m)$ と、 m_2 番目の状態から m 番目の状態への遷移に伴う枝尤度 $m_b(m_2, m)$ を読み出し、式(5)に示した m_o を計算する。

【0035】式(5)は、書きかえると、遷移後の最尤状態の状態尤度 $m_{s1} = m_s(m_1) + m_b(m_1, m)$ と、最尤状態との比較で選択されなかったパスの状態尤度 $m_{s2} = m_s(m_2) + m_b(m_2, m)$ の差となり、これが復号出力の信頼度、すなわち、軟判定出力尤度情報となる。最尤復号器では、生き残りパスが1本になった時点で最尤復号が可能となるので、ACS演算部2が選択情報を出力してから、最尤復号部6が最尤判定するまでの間、選択情報をパスメモリ5に記憶しておく。同様に、出力尤度演算部80が出力する軟判定出力尤度情報も出力尤度メモリ81に記憶しておき、最尤復号部6が最尤判定した時点で、最尤復号出力に対応する軟判定出力尤度情報を出力尤度メモリ81から読み出して出力する。以上述べた本発明による第2の実施例によって、軟判定尤度情報を出力する再帰形組織畳み込み符号用の最尤復号器が

構成できる。

【0036】

【発明の効果】以上説明したように、本発明による最尤復号器によれば、再帰形組織畳み込み符号用の最尤復号器で、軟判定尤度情報を出力するように構成することができる。硬判定出力形の再帰形組織畳み込み符号用復号器では、従来の方法でも構成可能であるが、この場合でも、符号除数多項式の乗算回路を省略することができるので、回路規模の縮減、演算処理時間の短縮が図れる。さらに、回路の切り替えにより、再帰形組織畳み込み符号に対応する部分を用いなければ、通常の前組織畳み込み符号用の最尤復号器として用いることもできる。

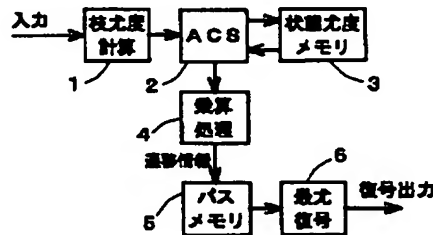
【図面の簡単な説明】

【図1】 本発明の最尤復号器の一実施例の構成を示すブロック図。

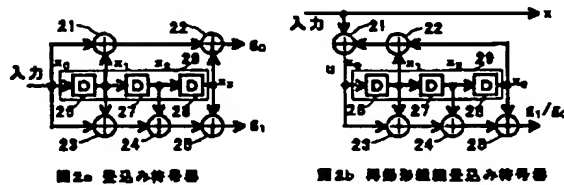
【図2】 従来の最尤復号器の動作を説明するための図。

【図3】 従来の最尤復号器の動作を説明するための畳み込み符号のトレリス線図。

【図1】



【図2】



【図 3】

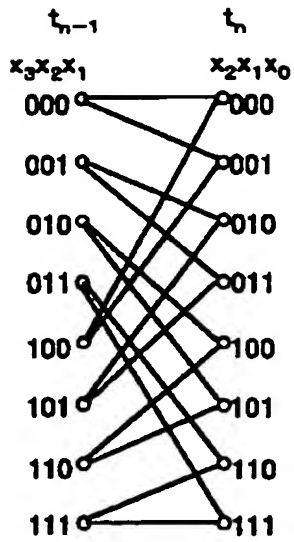


図3a 8状態トレリス

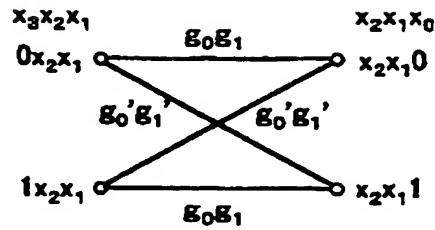
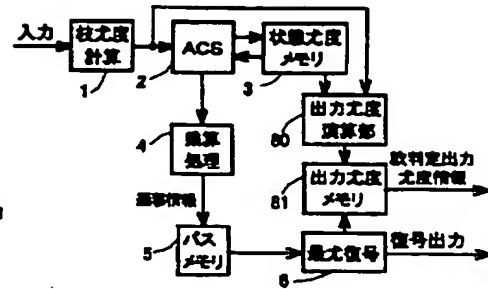
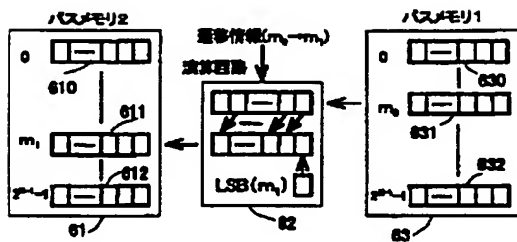


図3b トレリスの基本単位

【図 8】



【図 6】



【図 7】

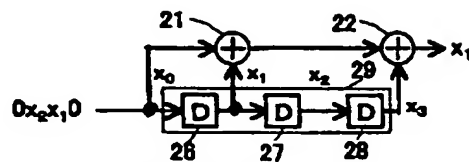


図7b 反帰回路

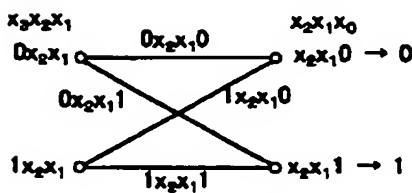


図7a (非組織) 畳み込み符号の場合

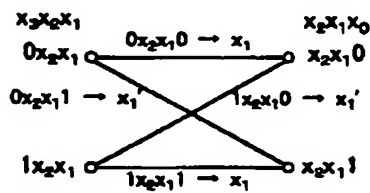


図7c 再帰組織畳み込み符号の場合